

[INICIAR SESIÓN](#)[NUESTROS PLANES](#)[TODOS LOS CURSOS](#)[FORMACIONES](#)[CURSOS](#)[PARA EMPRESAS](#)[ARTÍCULOS DE TECNOLOGÍA > PROGRAMACIÓN](#)

Recibiendo datos en Java



Alex Felipe

20/05/2022



```
Alex Felipe;21  
Cléber Nascimento;16  
Felipe de Souza;25  
João da Silva;18  
Juliana Guimarães;18  
Maria Cristina;23  
Pâmela Aguiar;17
```

Recibí una lista de estudiantes en un archivo CSV y necesito registrarlos en mi sistema:

```
Alex Felipe;21
Cleber Nascimento;16
Felipe de Souza;25
João da Silva;18
Juliana Guimarães;18
Maria Cristina;23
Pâmela Aguiar;17
```

```
public class Alumno {

    private final String nombre;
    private final int edad;

    public Alumno(String nombre, int edad) {
        this.nombre = nombre;
        this.edad = edad;
    }
}
```

Necesito ingresar la información de los alumnos en mi sistema. Usaré la clase Scanner del paquete java.util para recibir los datos del teclado:

```
Scanner sc = new Scanner(System.in);

System.out.println("Digite el nombre del alumno:");
String nombre = sc.nextLine();
System.out.println("Informe la edad del alumno:");
int edad = sc.nextInt();

System.out.println("alumno: " + nombre + "\\nedad: " + edad);
```

Obteniendo el primer resultado:

```
Digite el nombre del alumno: Alex Felipe
Informe la fecha de nacimiento del alumno: 21
```

```
alumno: Alex Felipe  
edad: 21
```

El método `nextLine()` permite la lectura de `String` y el método `nextInt()` permite la lectura de `int`. Envío de datos al alumno:

```
Alumno alumno = new Alumno(nombre, edad);  
  
System.out.println(alumno);
```

Verificando la impresión:

```
nombre: Alex Felipe edad: 21
```

¡Genial, logré registrar a un estudiante! Ahora vamos a registrar al resto de los estudiantes:

```
Digite el nombre del alumno: Cleber Nascimento Informe la edad del alumno: 16
```

...



¡Espera un segundo! ¡Esto ya está cansando! ¿Tendré que registrar a todos estos alumnos a mano? ¡Si con apenas 7 alumnos da mucho trabajo imagina con 30 más! ¡Acabará con la paciencia de todos!

¡Además de permitir la lectura de datos a través del teclado, la clase `Scanner` **permite la lectura de archivos!** Simplemente informe al constructor que, en lugar de leer desde el teclado (`System.in`), queremos leer un archivo específico instanciando la clase `File` del paquete `java.io`:

```
Scanner sc = new Scanner(new File("usuarios.csv"));
```

¡Listo! Ahora necesitaremos leer línea por línea iterando con un `while` a través del método `hasNext()` que informa si aún hay líneas para leer:

```
while (sc.hasNext()) {  
    //El while solo se detiene cuando no hay más líneas para leer.  
}
```

Para obtener el siguiente elemento del archivo, usamos el método `next()`:

```
while (sc.hasNext()) {  
  
    String elemento = sc.next();  
    System.out.println(elemento);  
  
}
```

Si probamos nuestro código:

```
Alex Felipe;21  
Cleber Nascimento;16  
Felipe de Souza;25 ...
```

¡Ups! Así no podré separar los datos del alumno... Además de leer el archivo, el `Scanner` tiene el método `useDelimiter()` que **delimita la lectura del elemento**, es decir, determina hasta dónde necesita llegar el elemento, por ejemplo, quiero que tome el elemento hasta que aparezca un `;`:

```
Scanner sc = new Scanner(new File("archivo.csv"))  
    .useDelimiter(";");
```

Ejecutando el código una vez más:

```
Alex Felipe 21  
Cleber Nascimento 16  
Felipe de Souza 25  
  
...
```

¡Excelente! Separé la información, ahora instanciaré a mis alumnos:

```
String nombre = sc.next();  
int edad = Integer.parseInt(sc.next());  
  
Alumno alumno = new Alumno(nombre, edad);  
System.out.println(alumno);
```

Probando nuestro código:

```
Exception in thread "main"  
java.lang.NumberFormatException: For input string: "21 Cleber Nascimento"
```

Qué raro... ¿Problema de conversión? ¿Qué String es esta? Parecía que la información de nombre y edad estaba separada correctamente... Lo que sucedió en este caso es que nuestro delimitador está separando la información hasta el momento en que encuentra un ";" entre la edad y el siguiente nombre del estudiante no hay ";" tiene solo un salto de línea, es decir, debemos informarle al Scanner que **al mismo tiempo lo delimitamos por el ";" también lo queremos delimitado por un salto de línea:**

```
Scanner sc = new Scanner(new File("archivo.csv")).useDelimiter(";|\\\\\\n");
```

El pipe (|) indica que queremos que esté delimitado por ";" o "\\n" que significa salto de línea. Probando de nuevo:

```
Exception in thread "main"  
java.lang.NumberFormatException: For input string: "21 "
```

¿Otra excepción? Como podemos ver, ya separó la edad del nombre del otro estudiante, pero aún dejó un salto de línea y por lo tanto no fue posible convertir a int. En este caso, podemos utilizar el método `trim()` de la propia clase `String`, que ajusta el contenido contenido en el `String`, **eliminando saltos de línea o espacios en blanco:**

```
String nombre = sc.next();  
int edad = Integer.parseInt(sc.next().trim());  
  
Alumno alumno = new Alumno(nombre, edad);
```

```
System.out.println(alumno);
```

Dando el resultado:

```
nombre: Alex Felipe edad: 21  
nombre: Cleber Nascimento edad:  
16 nombre: Felipe de Souza edad: 25
```

...

¡Estupendo! ¡Ahora mi sistema puede leer el archivo CSV y registrar a los estudiantes por mí!

Vimos que para recibir datos de medios externos en Java podemos usar la clase Scanner que tiene la capacidad de recibir datos a través del teclado o archivos, pero no siempre cuando leemos información de un archivo se espera el resultado, por lo que siempre debemos verificar la siguiente información:

- Si hay algún carácter que separa cada elemento, use un delimitador
- Si necesita usar más de un delimitador, use pipe (|)
- Si siempre obtiene el siguiente elemento en lugar de la línea completa, use siempre el delimitador '\n'

¿Te gustó la clase de Scanner? ¿Quieres conocer otras alternativas a la clase Scanner y otras bibliotecas del lenguaje Java? Conoce la [Formación en Java](#).



Alex Felipe Victor Vieira

Alex es instructor y desarrollador y tiene experiencia en Java, Kotlin, Android. Creador de más de 40 cursos como Kotlin, Flutter, Android, persistencia de datos, comunicación de API web, personalización de pantalla, pruebas automatizadas, arquitectura de aplicaciones y Firebase. Es experto en Programación Orientada a Objetos, siempre con el objetivo de compartir las mejores prácticas y tendencias en el mercado de desarrollo de software. Trabajó durante 2 años como editor de contenidos en el blog de Alura y hoy sigue escribiendo artículos técnicos.

Este artículo fue adecuado para Alura Latam por: [Jose Charris](#)

Cursos de Programación

ARTÍCULOS DE TECNOLOGÍA > PROGRAMACIÓN

**En Alura encontrarás variados cursos sobre Programación.
¡Comienza ahora!**

SEMESTRAL

US\$49,90

un solo pago de US\$49,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana

- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 6 meses

¡QUIERO EMPEZAR A ESTUDIAR!

[Paga en moneda local en los siguientes países](#)

ANUAL

US\$79,90

un solo pago de US\$79,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas



Acceso a todo el contenido de la
plataforma por 12 meses

¡QUIERO EMPEZAR A ESTUDIAR!

[Paga en moneda local en los siguientes países](#)

Acceso a todos
los cursos

Estudia las 24 horas,
dónde y cuándo quieras

Nuevos cursos
cada semana

NAVEGACIÓN

PLANES
INSTRUCTORES
BLOG
POLÍTICA DE PRIVACIDAD
TÉRMINOS DE USO
SOBRE NOSOTROS
PREGUNTAS FRECUENTES

¡CONTÁCTANOS!

¡QUIERO ENTRAR EN CONTACTO!

BLOG

PROGRAMACIÓN
FRONT END
DATA SCIENCE
INNOVACIÓN Y GESTIÓN
DEVOPS

AOVS Sistemas de Informática S.A
CNPJ 05.555.382/0001-33

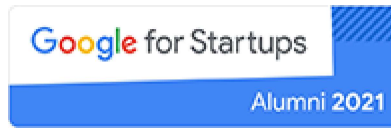
SÍGUENOS EN NUESTRAS REDES SOCIALES



ALIADOS



En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

CURSOS

Cursos de Programación

Lógica de Programación | Java

Cursos de Front End

HTML y CSS | JavaScript | React

Cursos de Data Science

Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

Cursos de DevOps

Docker | Linux

Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics | Liderazgo y Gestión de Equipos | Startups y Emprendimiento